UNIVERSITY OF SOUTHAMPTON

Applying Gamification to Teaching Cyber Security

By

Reece Buckle

Project Supervisor: Dr Nawfal Fadhel

Second Examiner: Dr Andrew Sogokon

A project progress report submitted for the award of BSc Computer Science

Department of Electronics and Computer Science

Novemeber 2020

Abstract

Due to the ever changing landscape of technology and cyber security, there is a requirement to promote cyber security awareness where possible. Typically this is done via government endorsed training schemes, website campaigns, fliers and posters and gamified strategies. As a strategy towards this problem, this report investigates the use of gamification mechanics in order to teach cyber security concepts in a fun and interactive way. Furthermore, despite many serious cyber security games being developed for Universities and businesses, there are a lack of cyber security games that are also relevant and available to the general public. Therefore, this project will aspire to develop a multiplayer board game in which cyber security vulnerabilities are illustrated through mini-game challenges, cyber attack & defence cards and a contextually appropriate setting. The goal of this project is to produce a game which is relevant for both businesses and the public domain - and ultimately leaves the player more conscientious in regards to how they interact with technology.

1 Statement of Originality

I have read and understood the ECS Academic Integrity information and the University's Academic Integrity Guidance for Students.

I am aware that failure to act in accordance with the Regulations Governing Academic Integrity may lead to the imposition of penalties which, for the most serious cases, may include termination of programme.

I consent to the University copying and distributing any or all of my work in any form and using third parties (who may be based outside the EU/EEA) to verify whether my work contains plagiarised material, and for quality assurance purposes.
I have acknowledged all sources, and identified any content taken from elsewhere.

I have not used any resources produced by anyone else.

I did all the work myself, or with my allocated group, and have not helped anyone else.

The material in the report is genuine, and I have included all my data/code/designs.

I have not submitted any part of this work for another assessment.

Contents

List of Figures

2 Introduction

2.1 Problem Summary

Problem Statement

Despite the existence of many cyber security awareness programs, there is still a lack of effective, widespread cyber security training

As modern-day technology is ever evolving, the number of users who interact with technology on a daily basis increases consequently. As a result, the risk of an individual, or business, becoming a victim to cyber-crime increases proportionately. In particular, small and medium-sized businesses (SMBs) are the biggest sectors targeted by cyber-criminals [3], which stem from issues such as budget restraints and expressing a lack of understanding towards cyber security concepts.

In fact, as a consequence of COVID-19 changing the dynamic of industry standards this year, a statistical analysis from May 2020 (UK) showed that individuals experiencing targeted hacking increased by 77.41% - in comparison with the previous year [4]. This is most likely due to the fact that employees are encouraged to work from home via their personal computers. Consequently, this has fed into a new strategy whereby cyber-criminals are moving laterally into organisational infrastructure by targeting and infecting employees at their less secure personal computers [5].

In regards to this problem, this paper will explore the effectiveness of educational games - which has been shown to have an advantage on the learning outcome in comparison with traditional training material [6]. Therefore, this paper presents the following research question and hypothesis:

Research Question

Does teaching cyber security through a gamified medium improve user confidence in protecting against cyber attacks?

Hypothesis

Creating an educational cyber security game will leave users feeling more confident and aware with regards to cyber security concepts

2.2 Goals & Scope

The goal of this project is to investigate how to effectively apply gamification mechanics in order to teach cyber security principles appropriately. The expected result of this project is to create a multiplayer, online tabletop board game.

3 Literature Review & Research

3.1 Introduction

To date, this report includes an evaluation of literature review pertaining to pre-existing cyber security games, mechanics for game-based learning and current trends in cyber security training methodologies.

3.2 The Problem with Current Cyber Security Training Programs

This report will specifically analyse the shortcomings and difficulties that relate to current training programs designed for small and medium-sized businesses [3, 7].

Problem	Description
1	SMBs can be heavily constrained by a limited budget
2	SMBs can be difficult to reach as they do not understand the severity of data breaches
3	SMBs are often distracted by the operational requirements for setting up and running a small business
4	SMBs struggle to identify their assets in terms of the risks associated with them

In regards to the delivery of training programs, providing generalised cyber security advice (from an independent advisor) has been shown to have little effect on changing the behaviour of employees within SMBs [3, 8]. Furthermore, traditional training simulations (including gamified video simulations) are generally undertaken in a formal environment which leads to a situation of recipients not absorbing the information well [2].

The methodology of cyber security exposure is also important as whilst employees may understand some general information about the vulnerability demonstrated, they may still fail to see how it relates to their workplace environment [7] or how they link together in a multifaceted social engineering attack [8]. This last point emphasises the need for widespread conceptual training in cyber security.

3.3 A Critical Analysis of Pre-Existing Cyber Security Games

For a full account of educational games and resources reviewed, please see Appendix A. My methodology for reviewing cyber security games was two fold:

- First - utilising the Google search engine with the following keywords: ‘cyber security’, ‘serious games’, ‘gamification’ and ‘game-based learning’ in order to look for any widely available games. These commonly returned web applications designed for students in all stages of education.

- Second - utilising Google Scholar and the IEEE Database with similar keywords in order to find academic papers which either reviewed other educational games, or were proposing one. For the games that were not available online, I summarised the key information and research results from the academic source material.

Building on from this, I concluded the following categories of educational game-based learning strategies in order to identify the most appropriate medium for the purpose of answering the research question and hypothesis established in this project.

3.3.1 Web Applications

Advantages	Disadvantages
Simple point and click interactivity	Can lack depth and relevance to a specific target (often designed for students)
Easily accessible anywhere with an internet connection	Not suitable for offline usage
Cheap development cost & time	

3.3.2 Video / Simulation

Advantages	Disadvantages
Contextually appropriate for use within the workplace [7]	Not appropriate for the general public & students
Accessible both online & offline	Requires multiple play-throughs if scenarios are divergent
	Typically not very fun as undertaken in a formal environment [7]

3.3.3 Cooperative Tabletop

Advantages	Disadvantages
Encourages social engagement and team-working	Requires multiple players
Cheap to prototype and produce a physical implementation	Requires much fine-tuning of rules and mechanics implemented
Encourages thinking strategically	

3.3.4 Task Management

Advantages	Disadvantages
Easy to employ around current learning strategies (within the classroom or workplace)	Requires long term evaluation of effectiveness
Perfect example of procedural learning [9]	Not a true application of an educational game

3.3.5 Single Player

Advantages	Disadvantages
Immersive and engaging typically through story driven content	High development cost & time
Often places the player as a white / black hat hacker which encourages adversarial thinking	Lack of exploration on how to prevent vulnerabilities as a target

3.4 Difficulties that Pre-Existing Cyber Security Games Face

Many of the educational games reviewed relied heavily on presenting facts and then subsequently quizzing the user with a related question. However, users can utilise common sense to rule out incorrect answers thus fail to invoke critical thinking and do not keep the user engaged.

As a solution, gamified strategies should incorporate a variety of factual, conceptual and procedural learning methodologies [9]. In particular, a strong conceptual understanding should be prioritised due to the rapidly changing landscape of cyber security in which cyber-criminals will always be engineering new attack vectors [8]. Therefore, it is imperative for end-users to be able to adapt their way of thinking when interacting with new technology.

For procedural learning, both Nova Cyber Lab [10] and Classcraft [11] (Appendix A) exemplify this by beginning with simple challenges and progressively increasing the difficulty of said challenges as the user progresses. Unlike the other games reviewed, Classcraft is unique as it encourages users within a team to continually expand upon their knowledge by working towards new goals and objectives collaboratively. Furthermore, this system incorporates real-world rewards and punishments to encourage user-engagement.

Finally, many serious cyber security games are designed primarily for university students and businesses but are not readily available to the general public [9]; this agrees with my own research - whereby many of the publicly accessible educational games I reviewed were considerably outdated and not intended for the general public.

3.5 Why Use a Game-Based Learning Approach?

As identified above, many challenges that arise when attempting to educate employees within the business environment occur due the formality of traditional training methodologies in which employees will aim to complete as fast as possible [8]. This absentmindedness often results in training content which is not absorbed effectively [3, 8]. However, these challenges can often be overcome with game-based learning strategies [6].

A summary of key strengths and motivations for gamification [6, 12, 13, 14, 15] include:

- Rapid progress paired with instant feedback
- Strong user engagement
- Allows the user to learn from mistakes in a safe, non punishing environment which would otherwise discourage exploration (due to the fear of failure)
- Serves as a platform to encourage self-learning at the user's desired pace
- Allows the user to become deeply immersed into the 'game - world' where learning feels like a secondary objective
- Relatively cheap and low cost to produce in comparison to larger training schemes
- Requires little to no supervision
- Easy to distribute across multiple platforms and incorporate into the workplace
- Easy to integrate within events such as hackathons and other cyber security awareness gatherings
- Use of an integrated rewards system such as badges, hidden achievements and the desire to win
- Educationally appropriate by incorporating a structure/story that is contextually similar to the real world

3.6 Appropriate Gamification Mechanics

Forde's report [16] identifies the following gamification mechanics in order to increase the adoption rate of effective cyber security standards within the workplace:

- Avatar / User Profile
- Badges / Privileges
- Challenge
- Competition
- Collaboration
- Feedback / Guidance
- Goals / Objectives
- Incentives / Rewards
- Leaderboards
- Points System
- Progress / Levels
- Role Playing
- Story
- Tips / Hints System

3.7 Conclusion

To summarise, cyber security training programs need to be cost effective and target the specific needs and requirements of the workplace in order to be viable for SMBs. Furthermore, where traditional training methodologies fall short, gamified strategies can offer a more cost-effective solution for teaching relevant cyber security concepts. However, unless the game in question has a strong story (e.g. Hacknet [17]), many educational games reviewed included a number of flaws which stem from lack of gamified mechanics that encourage meaningful play. Forde's report [16] identifies these key mechanics as multiplayer, leaderboards, points system and competition which can be deemed as the most effective in motivating people to participate and learn about a subject that they would otherwise have little interest in. Lastly, there is a gap in the public domain in which an appropriately designed cyber security game could be of use.

4 Requirements for Project

Have re-written/updated all but the last two subsections for this section, just need to copy over and format nicely (ran out of time)

4.1 Introduction

4.2 Stakeholder - Personas

4.3 Functional Requirements

4.4 Non - Functional Requirements

4.5 User Stories

Was debating whether to do a product backlog here, but the Gantt charts serve basically the same purpose

4.6 Project Management Tools & Techniques

4.7 Constraints

4.8 Risk Assessment

Haven't updated risk assessment or gantt chart since last report - are these still necessary in this section?

4.9 Gantt Chart

5 Design of the Project

5.1 Introduction

5.2 Applying the MDA Framework

The MDA (Mechanics, Dynamics, Aesthetics) framework [18] is the most popular and well-known framework to identify the requirements of building a game. It is an iterative process in which the mechanics of the game are first outlined. These mechanics are then evaluated to see what game-play loops (dynamics) emerge. Finally, the dynamics are evaluated to choose a theme – e.g. the preservation of resources is an important dynamic for a survival title.

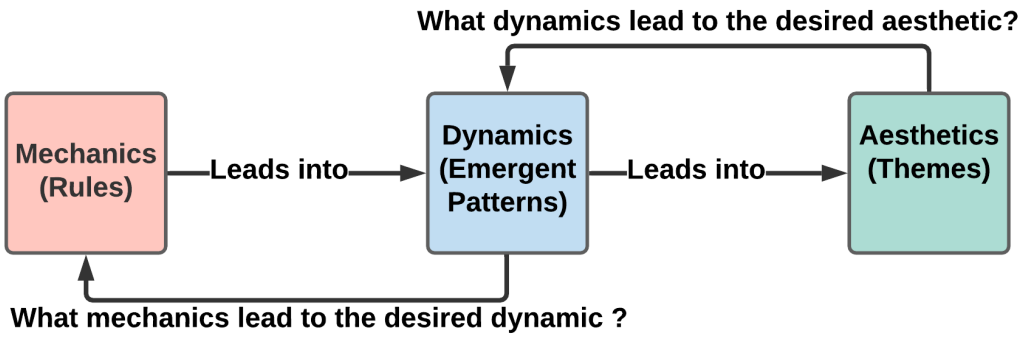

Figure 1: The Mechanics-Dynamics-Aesthetics Model

5.2.1 Mechanics

The mechanics are fundamentally the most basic rules and actions a player can do within the game world. They can be identified as follows:

- Action Point (AP) requirement for certain games
- Moves limited by a distance range and can be blocked by other
- Movement opportunities is blocked by both enemy and friendly units
- Units can only attack once and move once per turn
- Units can do damage, bypass shields, restore defence and disable other units
- Disabling a database / web server unit will stop AP gain from that unit for one turn
- Disabling movable units will prevent from acting for one turn
- Units have a max defence threshold cannot be restored past the threshold

- When a unit has 0 HP, it will be removed from the game world
- Units cannot move after attacking (even if they did not move that turn)
- Each movable unit has 2 abilities, whereas the static units (database / web server) cannot directly be interacted with

5.2.2 Dynamics

The dynamics can be thought of as 'the emergent behaviour that arises from gameplay when the mechanics are implemented' [18].They can be identified as follows:

- Players may choose to target the web server/database first to reduce AP gain
- Players may use shield destroying moves on heavy units
- Players may use cheaper moves to save AP
- Players may block their database/web server to protect them
- Players may use the bypass defence move to target units with a lot of defence but little HP (e.g. web server)

5.2.3 Aesthetics

The aesthetics can be thought of as 'the emotional response a game should illicit from the player' [18]. They can be broken down into :

- Sensation (emotion-invoking)
- Fantasy (immersion)
- Narrative (story - rich)
- **Challenge (puzzles / obstacles)***
- **Fellowship (co-operative / multiplayer)***
- Discovery (open world)
- Expression (character creation)
- Submission (simulations)

* Where Fellowship and Challenge are the two key target Aesthetics for the purpose of this game.

5.3 UML Modelling - A Class Diagram Representation

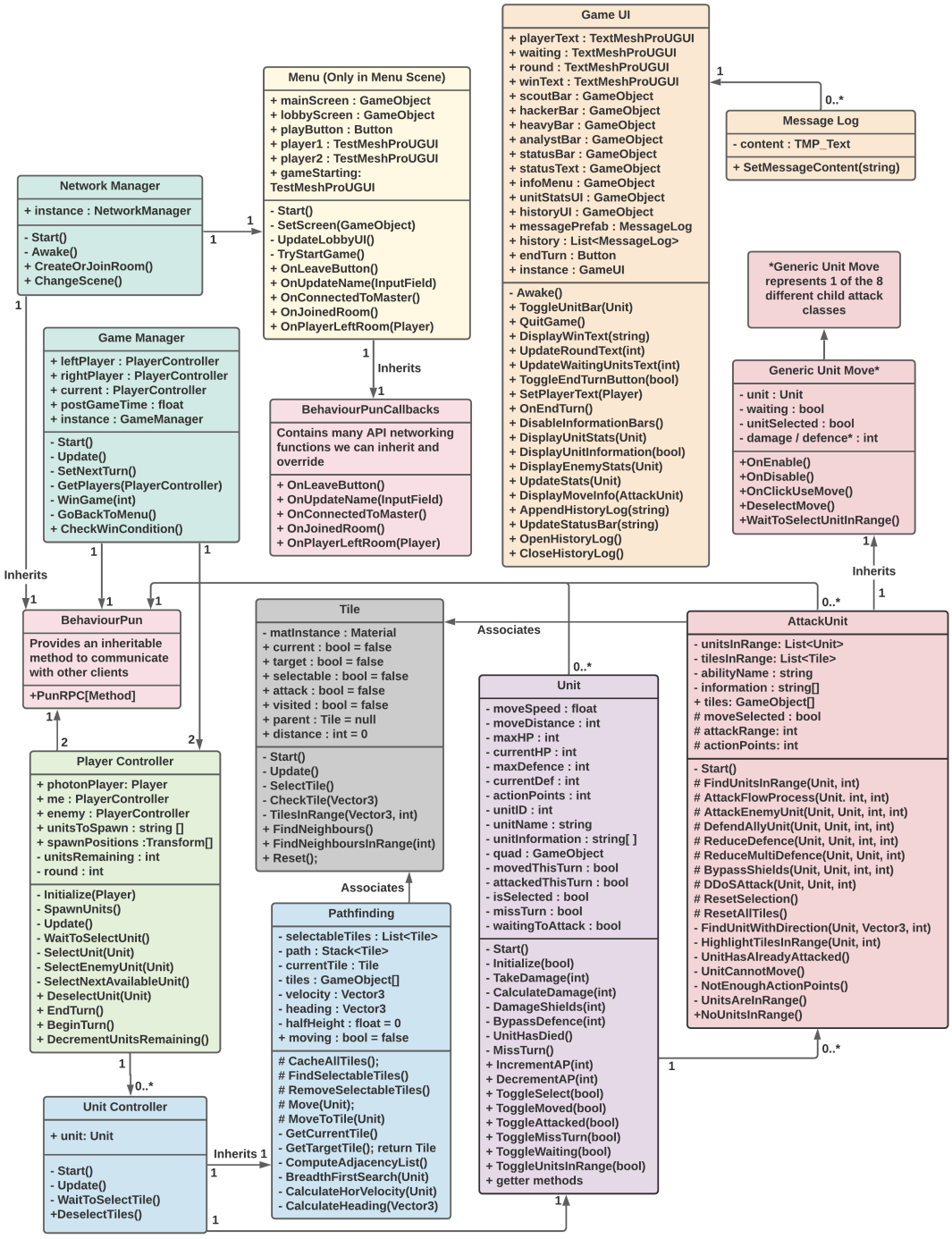

Figure 2: A Class Diagram Representation

5.4 Mapping the MDA Framework to Class Functions

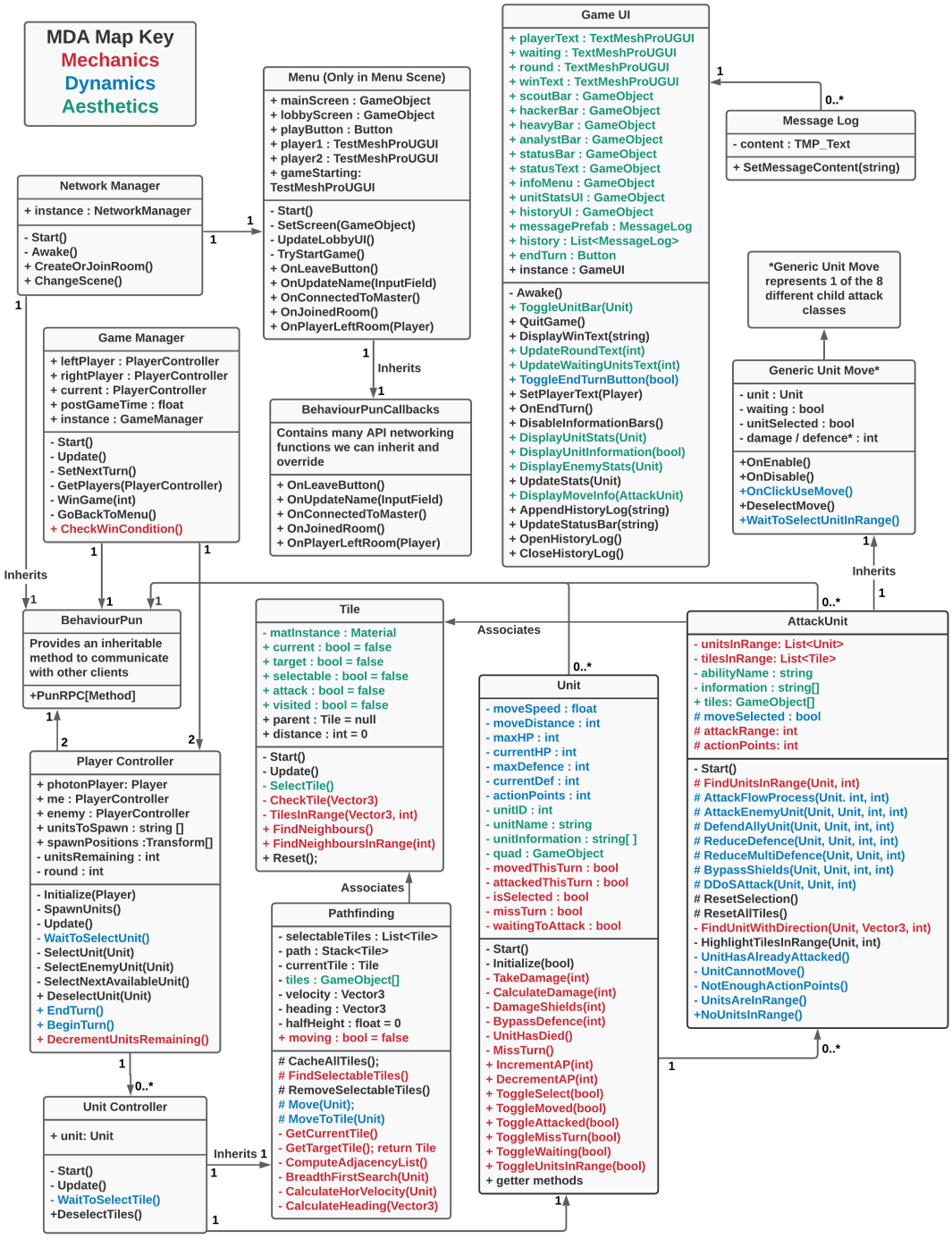

Figure 3: Mapping the MDA Framework to Class Functions

5.5 UML Modelling - Activity Diagram

The following Diagram refers to the sequence of two players joining a game, matchmaking and then the typical sequence of events that happens until the game is finished.

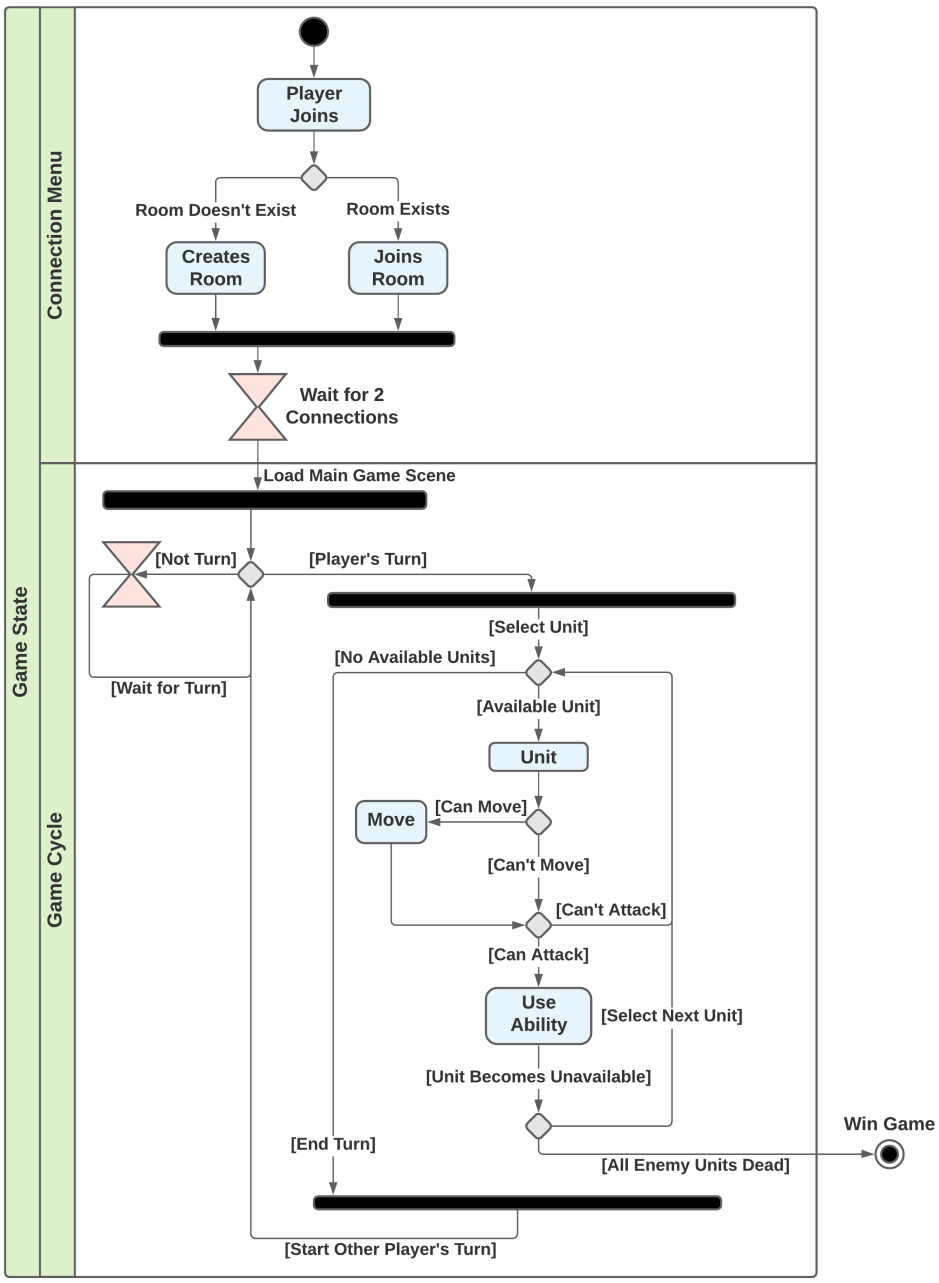

Figure 4: Illustration of a Typical Match

5.6 Visual Design, Contrast and Accessibility

The main UI colour was inspired from NieR:Automata. The hex values of these colours were then entered into a colour checker [19]to find an appropriate colour palette with the following complimentary and harmonic colours.

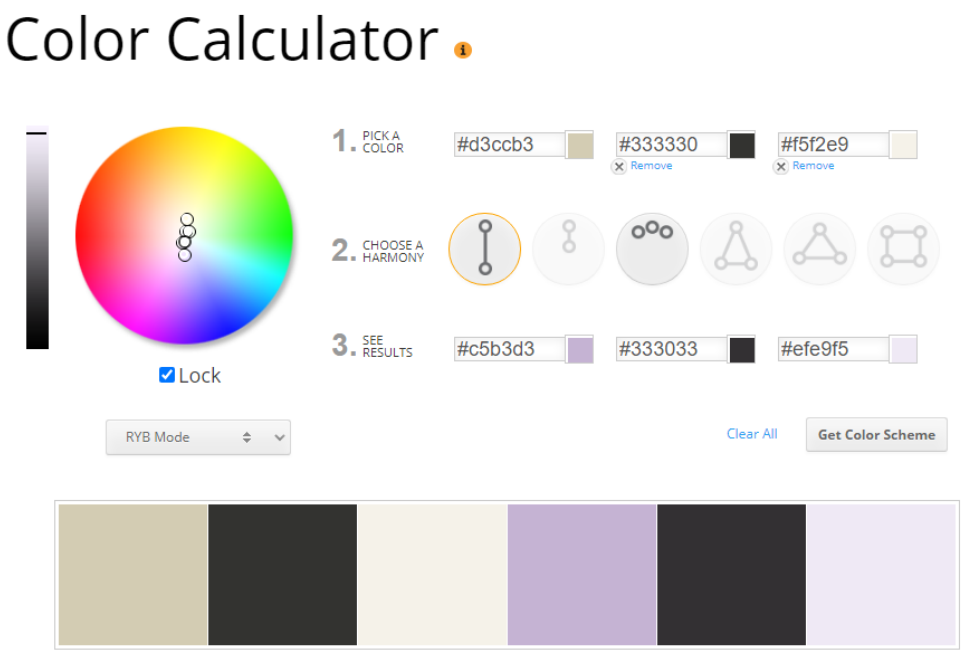

Figure 5: Identifying Complementary and Harmonic Colours

With regards to choosing an appropriate font size and style, the Web Content Accessibility Guidelines (WCAG 2.1) [20] states that the smallest text needs a minimum contrast ratio of 7:1.All text used within the application has a contrast ratio of 18.75:1 and 13.05:1 which is compliant.

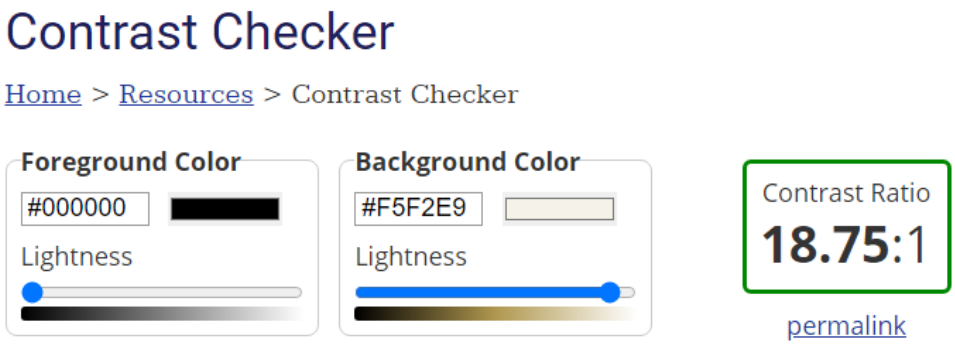

Figure 6: Contrast Checker for Larger Text [1]

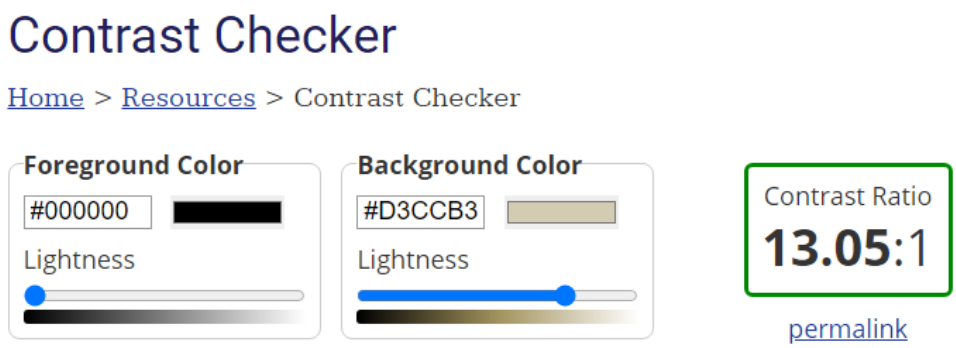

Figure 7: Contrast Checker for Smaller Text [2]

5.7 Colour Blindness

Approximately 1 in 12 men (8%), and 1 in 200 woman (0.5%) experience some form of CVD (Colour Vision Deficiency) [21]. It is therefore clear how important it is to design an application with accessibility in mind.

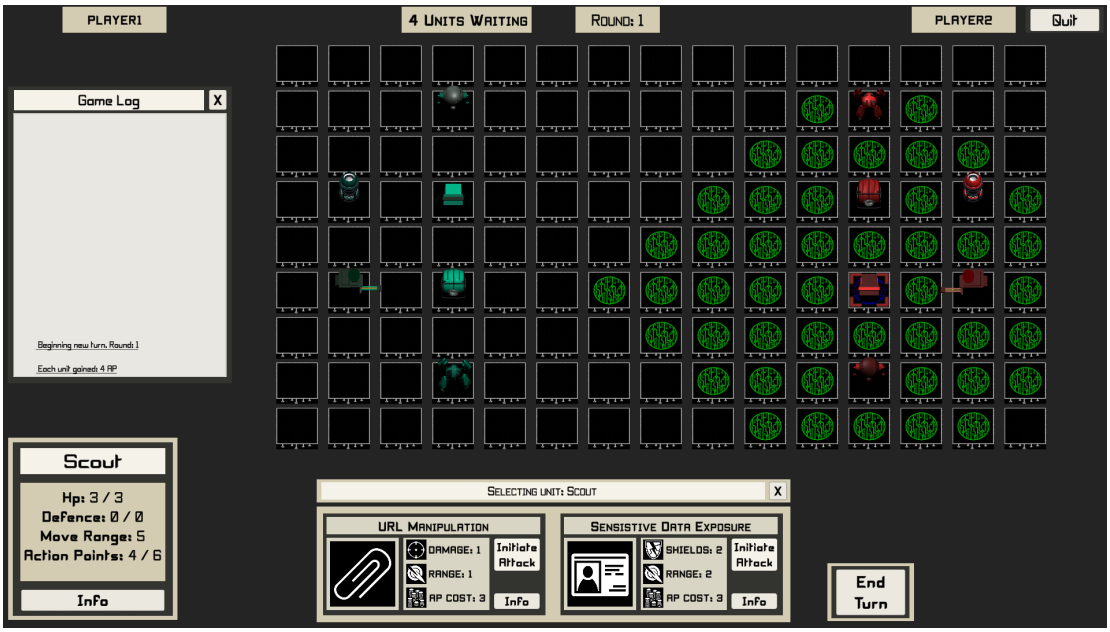

Figure 8: Colour Scheme before Revising to Accessibility Guidelines

With reference to the following Colour Blindness Guide [22], a colour-blind friendly colour palette was chosen and utilised for appropriate unit colours (deliberately avoiding the classic red and green which was initially used). As recommended within this guideline, a background pattern was also generated using a royalty-free pattern generator [23].

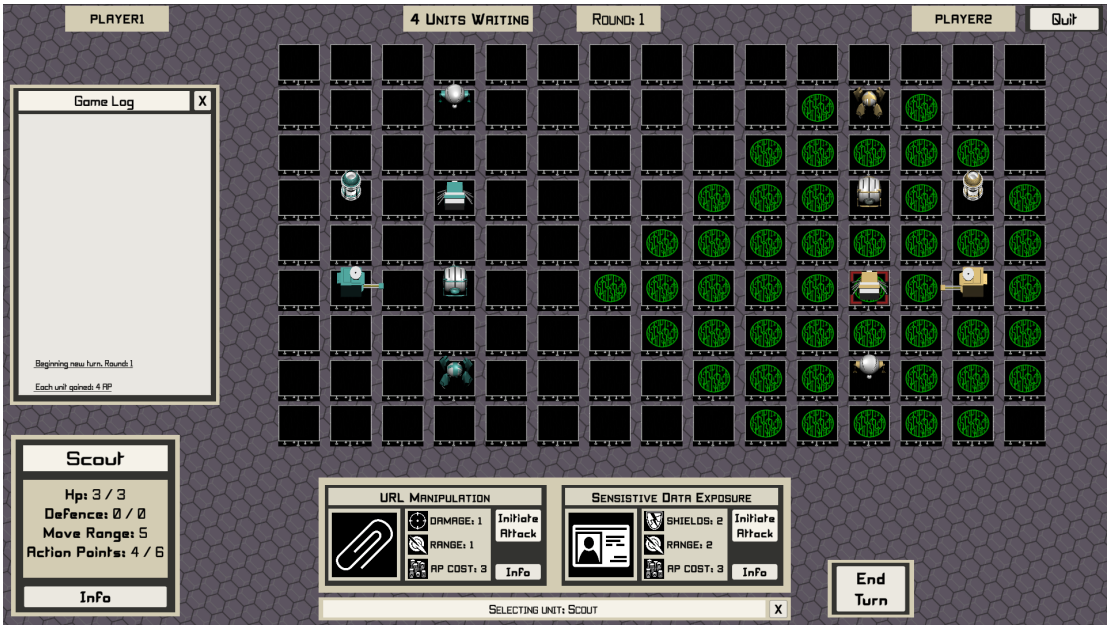

Figure 9: Revised Colour Scheme of the Windows - Edition

Due to some technical limitations when developing for the web, WebGL and Google Chrome proved to have many restrictions with rendering shaders, so an alternative version of the application had to be developed (with simpler block-coloured tiles). This can be seen below.

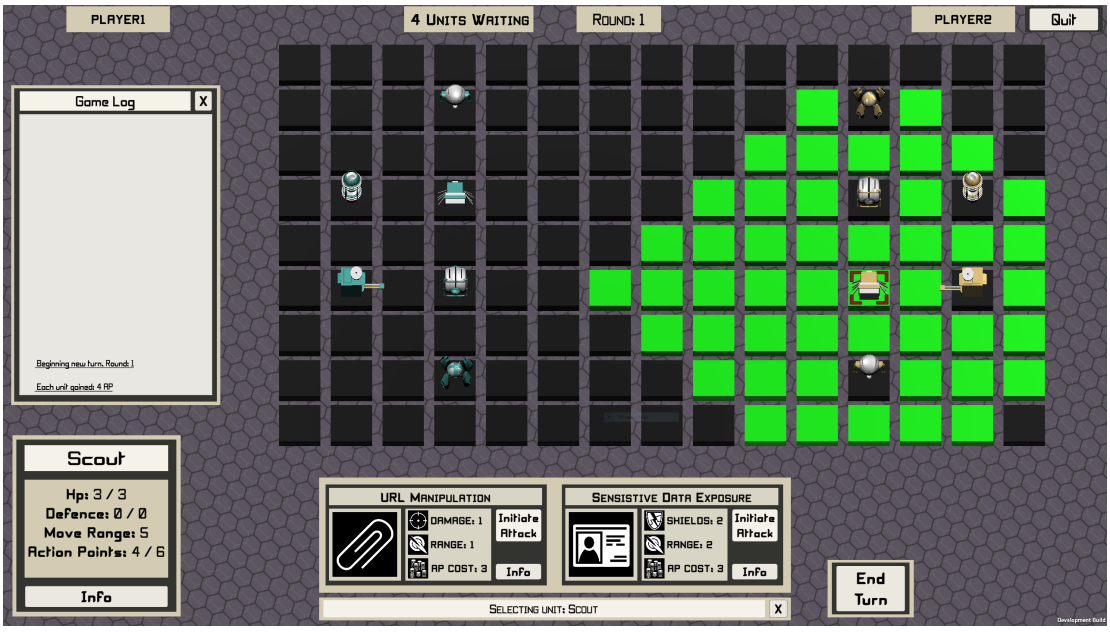

Figure 10: Revised Colour Scheme of the Web - Edition

5.8 Mapping the OWASP #10 to Game Mechanics

Need to copy OWASP attack information expressed through game mechanics here

5.9 Design Patterns (Unity C / C#

Integrates parts of all of these, need to link it to the example in the code

5.9.1 Singleton Design Pattern

A singleton involves creating a public static instance of a class or object such that only one version/reference of the object exists at any given time. This is very important for managers (such as the game manager, sound manager and player controller – as we do not need multiple instantiations of these objects, just one manager which can be statically called from other classes to invoke changes in the game's life-cycle)

5.9.2 Observer Design Pattern

The observer design pattern involves creating "listener" objects/event system which can respond to game interactions and invoke a method based on a change. For this project, the observer pattern was implemented within the UI / moves such that upon a move selected or used, the move will find the currently selected units, calculate potential targets in a nearby radius and update variables pertaining to what should be selected/highlighted – these are all temporarily cached, and then the move simply awaits a user input.

Though technically, we do not register for an event when OnEnable / OnDisable is invoked, so not a true implementation of the observer pattern.

5.9.3 Command Design Pattern

This pattern is extremely game-type specific and is an appropriate choice for turn – based strategies as it makes it possible to queue up actions, and reverse actions.

In essence, the player controller listens for human interactions and selects units which tells the unit controller (specific for each unit) that that unit is selected and available to move and use a move.

Furthermore, the moving itself involves a queue of tiles to movement and via the unit controller, processes each movement command until the unit has reached the target destination. More information is including about the movement later.

The command pattern makes it incredibly easy to implement reversal of movement, however I did not implement this feature in the game since I was designing a real time – synchronous multiplayer game (like Chess, where moves typically can't be revoked in the case of an accident). If this was a single player game, it'd be more appropriate to allow the player to reverse a move decision.

5.9.4 Component Design Pattern

This is the most common design pattern as Unity Development is hinged around components in the same way Java is hinged on objects. In essence, instead of having a single unit class which controls movement and control, the moves it has, the damage and stats, the buttons and UI that's linked to the unit, everything is broken down into sub-components and stitched together.

Typically a Unit object has a Unit Script (which contains it's unit information, health, defence and movement range), and a Unit Controller Script which can access it's unit information (movement range), but sole purpose is to calculate which tiles are movable by extending of the pathfinding class (which has all the movement operations and breadth first search). Unit moves (such as SQL injection) are completely decoupled from the Unit component, and are displayed by the game UI manager which a certain unit is selected, but these moves can be switched around and reassigned to any unit

5.9.5 Flyweight Design Pattern

This pattern is used to save memory when many instances of the same object are created and stored. For example, each time a tile changes colour, we access the material component and change the colour of the material. If we assigned this a new material colour, we could accidentally create a copy of the material where all the old colour materials will still be stored in memory (despite not even ever being reused or shown), which for a 100 tiles with 5 different colours would quickly lead to loads of excess tile-material objects existing. By using the profiler, we can access the material count and material memory used (testing). A better solution is we assign a single instance of a material for each tile (upon awake /object creation), and we update this instance (and not create a new material every time we change colour). Although this sounds like common sense, it is easy to miss!

6 A Justification of this Approach

Need to rewrite this section

7 Implementation of Project

7.1 Development Environment (Unity3D)

Unity3D [24] was chosen as an appropriate development environment because one of the key functional requirements identified was to design a game that could be exported to the web. Unity3D offers seamless integration with WebGL [25] to satisfy this requirement.

Phaser3 [26], HTML and NodeJS were considered as an alternative, web-development based solution. However, Phaser3 is still a relatively small open source project whereas Unity3D is a much more widely used industry trade skill. Furthermore, Unity3D has the largest online community, ample resources and integration for supporting frameworks.

7.2 Client - Server Multiplayer

Photon Cloud is a software as a service (SaaS) solution which provide, and host, cloud servers for free up until 20 concurrent players [27]. Beyond this, Photon offer paid services for more scalable solutions. Furthermore, Photon provides a free-to-use API, (Photon Unity Networking 2 [28]) which enable developers to integrate multiplayer functionality into their games. PUN2 was chosen because it has an ample tutorial base and very strong integration with Unity3D. Lastly, PUN2 supports cross platform gameplay which fulfils the requirement of being able to export and play on the web.

Mirror [29] is an alternative free and open source library which was built to replace Unity's previous deprecated multiplayer libraries (UNET). Although Mirror is completely free to use, relatively simple and highly covered, there were limitations in Mirror does not provide any functionality for externally hosting a game session. This would require developing, and running, a server lobby constantly to allow people to connect to the game. Clearly, this is would not be a viable long-term solution.

7.3 Photon Unity Networking 2 Architecture

PUN2 follows a classic Server – Client architecture. When a client connects to the Photon Network initially, it invokes the 'JoinOrCreateRoom()' function which essentially tries to join a non-full room (if it exists), or create one if there are no other rooms (with space) available.

The first player (client) to do this becomes the master client which is responsible for hosting the game. If the master client disconnects for any reason, there are options for the next available client (if present) to be promoted to a new master client. However, since this project is only designed for two players, it returns the other player to the main menu and ends the room session.

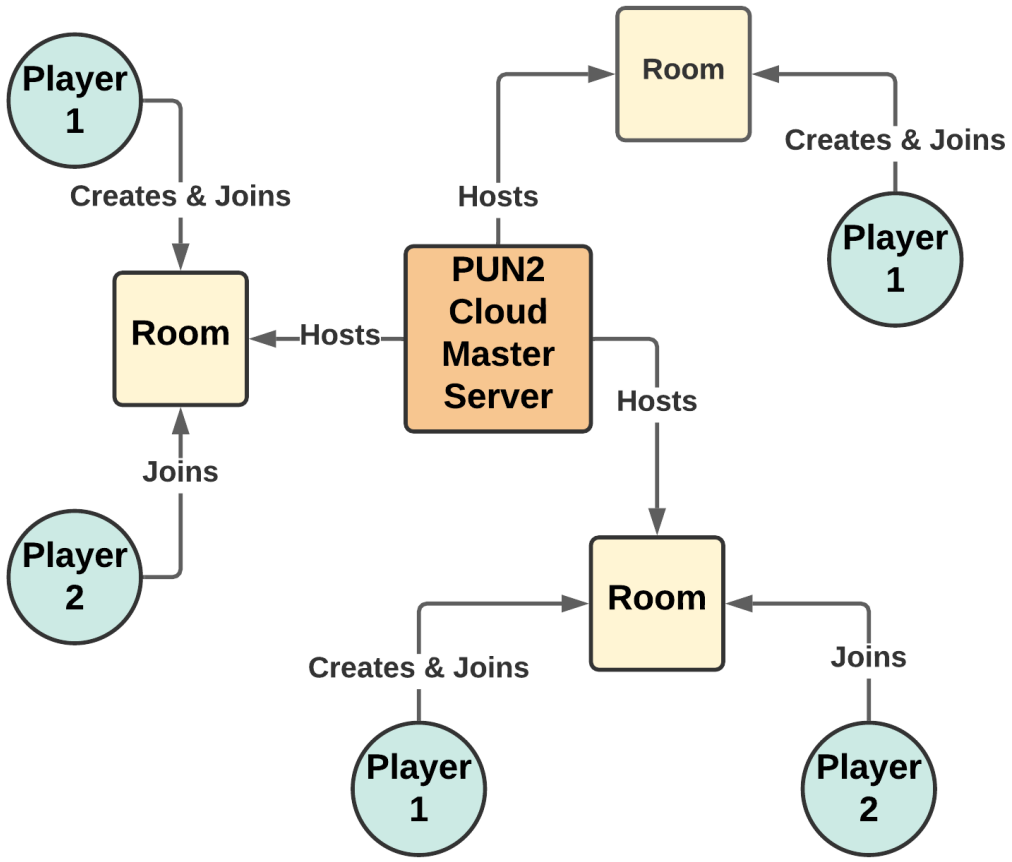

Figure 11: PUN2 Architectural Design

7.4 Remote Procedure Calls - RPCs

PUN RPCs (provided by the PUN2 API [30]) enable client - client communication in which method functions can be invoked on the opponents game version remotely. As both clients are running different, independent images of the game, RPCs are necessary to match these images and important state variables.

For example, after a unit has attacked another unit, it will remotely call the "Take Damage" function which PUN will invoke on the recipient client.

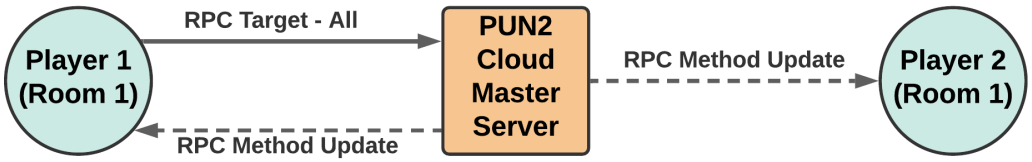

Figure 12: RPC Communication to One User

By calling 'RPC Target.All' however, the function is executed on all clients connected in the room (including the place of origin). This is the most often case since the majority of multiplayer functions are state-changing (such as beginning a new turn, and editing a units states), and these values must remain concurrent.

Figure 13: RPC Communication to All Users

7.5 Movement Mechanics - Breadth First Search (with Shortest Path)

To calculate the selectable tiles in range of a unit, Breadth First Search (BFS) was implemented due to its power to always return the shortest path.

When a unit is selected, a ray cast is sent below to find the current tile that the unit is standing on. From this tile, four more ray casts are emitted to discover any neighbours north, east, south, and west of the current tile; these tiles are added to an adjacency list. Any presently occupied tiles however are ignored as these are not valid tiles from which a unit can move to, or through.

BFS is only computed up until the maximum movement range specified by the Unit source.

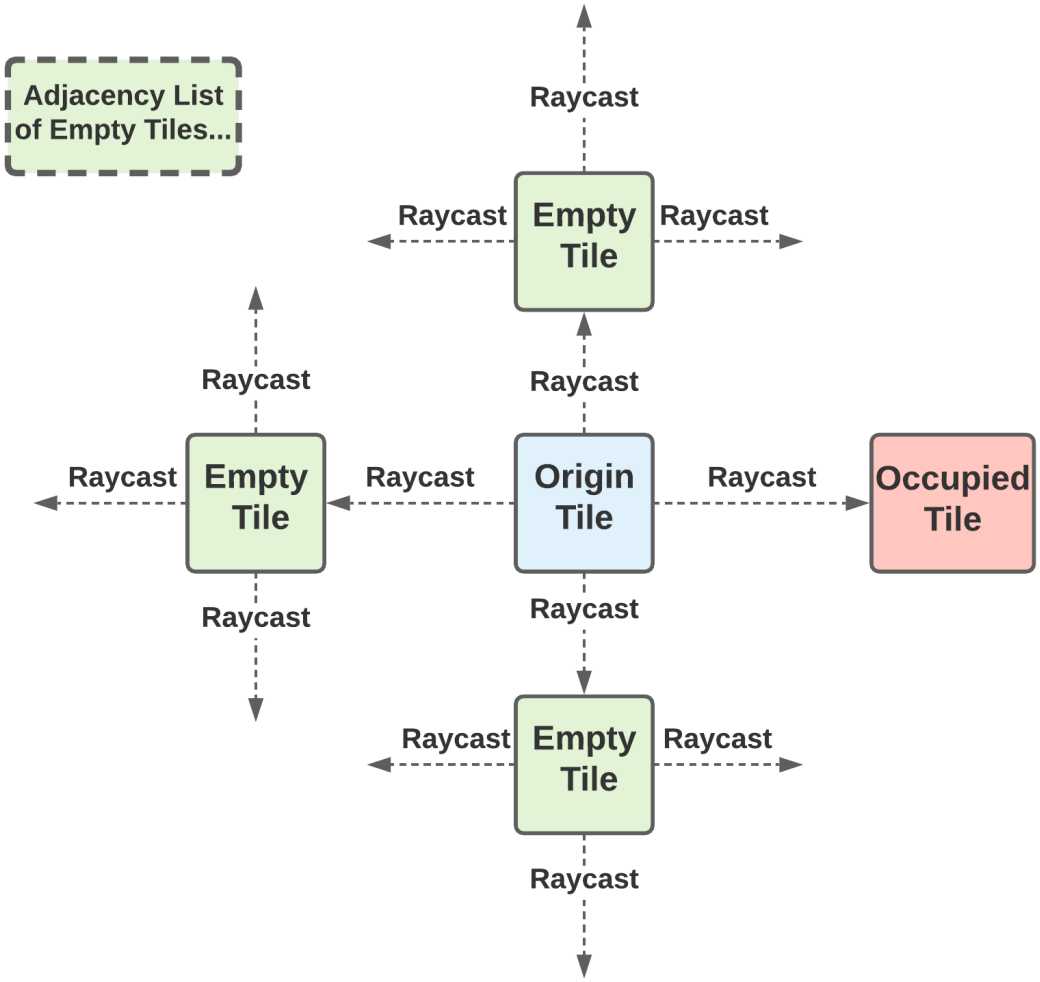

Figure 14: Ray-casting to Adjacent Tiles

BFS in this form utilises a queue data structure in which the current tile is enqueued and checked first, and then every adjacent tile thereafter. For each adjacent tile that is checked, if it is unoccupied, it is flagged as selectable and visited - such that it not checked twice. The diagram below illustrates the operation of checking tiles within a 2 – tile movement range.

If the queue ever reached 0, then the unit would effectively be in a position where it had no legal movies (i.e. it was surrounded by the edge of the map and blocked by other units).

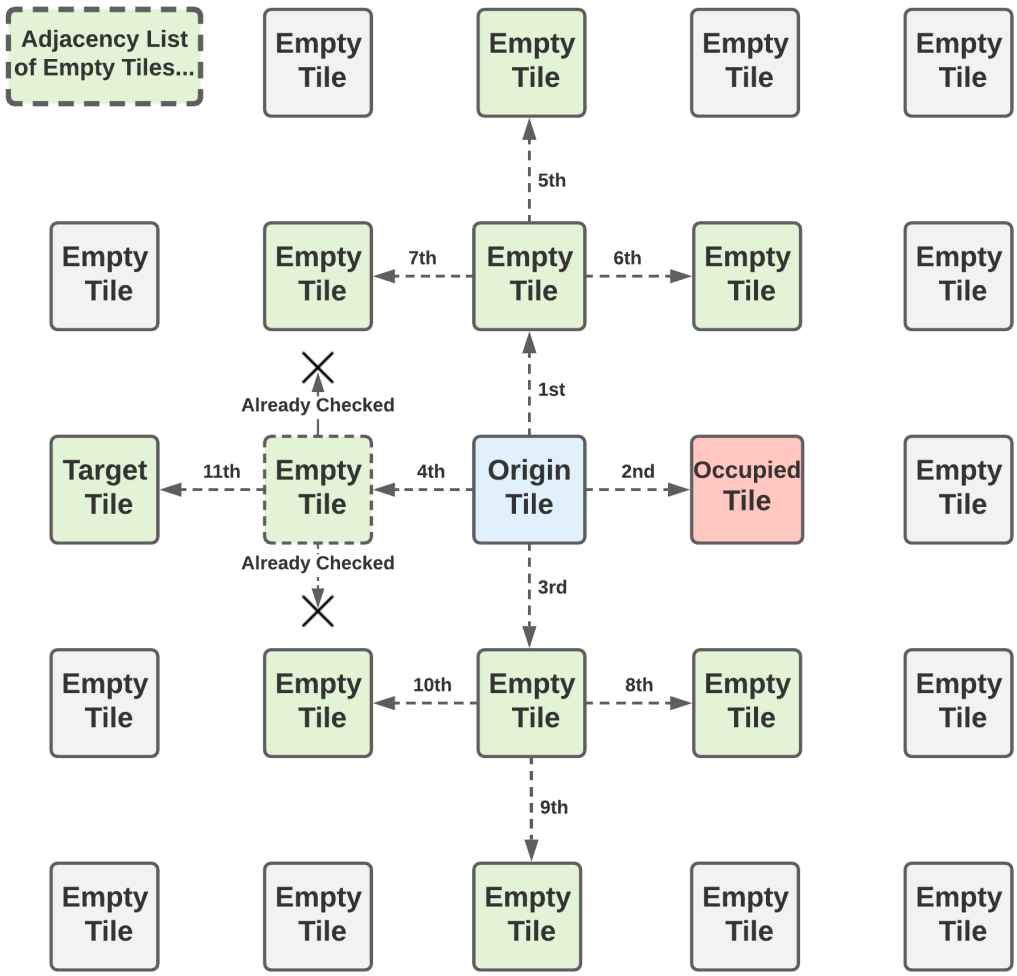

Figure 15: Finding Selectable Tiles via Breadth First Search

For each tile that is dequeued and checked, the parent tile is assigned to its child tile - such that when a user clicks on a selectable tile, we can access its parent recursively until we arrive back at the origin tile. This returns the discovered path to that tile (which is always the shortest path).

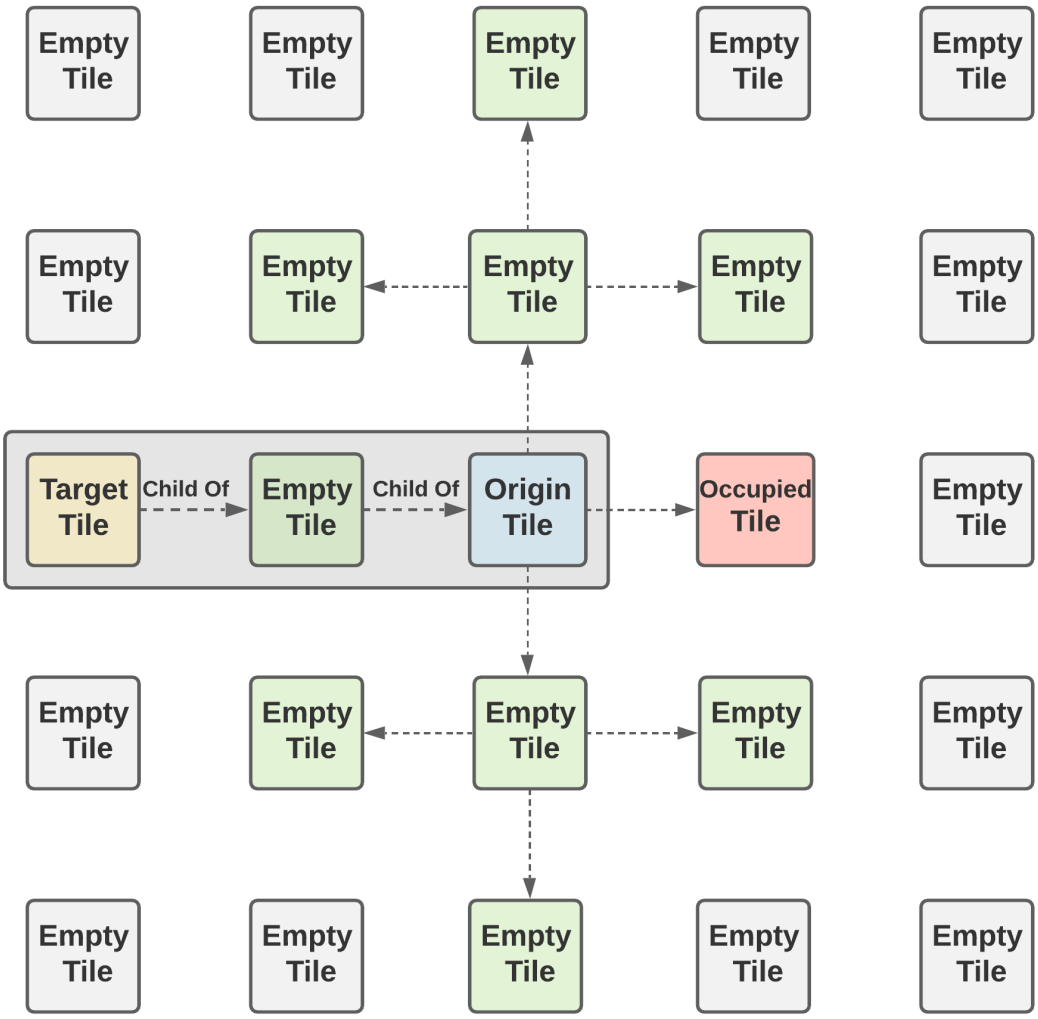

Figure 16: Finding the Shortest Path to Target Tile

Should I include some form of Pseudocode here?, or maybe write it out and add it in the Appendix?

8 Testing Strategy and Results

8.1 ParallelSync for Live Builds / Debugging

ParallelSync is a small open source extension for Unity3D which allows you to clone the Unity development environment and run multiple editors of the same project (which means you do not have to build and run the game every time to test 2 or more players!). This extension has proven to be an invaluable time saver for the development and quick testing of multiplayer development. ParallelSync works by cloning the structure of the project without needing to duplicate and load all files/assets/game states, instead it establishes pointers to the original files/assets and game scenes which have read access only (and do not affect the state of the master development environment). Because of this, it is easy to create, load and destroy as many clone environments as required.

8.2 Unit Test Cases

Completed but need to put in appendix/here and write up

9 Performance Testing

Completed, and mostly written up in word but still very messy! This will be a relatively long section though (lots of performance analysis done)

9.1 Integration Testing

10 A Critical Evaluation

Need to begin writing up

10.1 Evaluation of Approach

10.2 Evaluation of Final Implementation

10.3 User Feedback

(if time permits)

10.4 Evaluation of Networking Strategy

11 Conclusions and Future Work

11.1 My Contribution

(Began writing up in word, but still needs work)

11.2 Project Management and Reflection

11.3 Future Work

11.4 Final Conclusion

12 Bibliography

[1] 2021. [Online]. Available: https://webaim.org/resources/contrastchecker/?fcolor=000000&bcolor=D3CCB3

[2] 2021. [Online]. Available: https://webaim.org/resources/contrastchecker/?fcolor=000000&bcolor=F5F2E9

[3] M. Bada and J. R. Nurse, "Developing cybersecurity education and awareness programmes for small-and medium-sized enterprises (smes)," *Information & Computer Security*, 2019.

[4] D. Buil-Gil, F. Miró-Llinares, A. Moneva, S. Kemp, and N. Díaz-Castaño, "Cybercrime and shifts in opportunities during covid-19: a preliminary analysis in the uk," *European Societies*, pp. 1–13, 2020.

[5] H. S. Lallie, L. A. Shepherd, J. R. Nurse, A. Erola, G. Epiphaniou, C. Maple, and X. Bellekens, "Cyber security in the age of covid-19: a timeline and analysis of cyber-crime and cyber-attacks during the pandemic," *arXiv preprint arXiv:2006.11929*, 2020.

[6] J.-N. Tioh, M. Mina, and D. W. Jacobson, "Cyber security training a survey of serious games in cyber security," in *2017 IEEE Frontiers in Education Conference (FIE)*. IEEE, 2017, pp. 1–5.

[7] J. Abawajy, "User preference of cyber security awareness delivery methods," *Behaviour & Information Technology*, vol. 33, no. 3, pp. 237–248, 2014.

[8] H. Aldawood and G. Skinner, "Reviewing cyber security social engineering training and awareness programs—pitfalls and ongoing issues," *Future Internet*, vol. 11, no. 3, p. 73, 2019.

[9] R. Roepke and U. Schroeder, "The problem with teaching defence against the dark arts: A review of game-based learning applications and serious games for cyber security education." in *CSEDU (2)*, 2019, pp. 58–66.

[10] "Cybersecurity Lab - NOVA LABS," Accessed: 14-12-2020. [Online]. Available: https://www.pbs.org/wgbh/nova/labs/lab/cyber/

[11] "Classcraft," Accessed: 14-12-2020. [Online]. Available: https://www.classcraft.com/

[12] S. Hart, A. Margheri, F. Paci, and V. Sassone, "Riskio: A serious game for cyber security awareness and education," *Computers & Security*, p. 101827, 2020.

[13] A. Jøsang, V. Stray, and H. Rygge, "Threat poker: Gamification of secure agile," in *IFIP World Conference on Information Security Education*. Springer, 2020, pp. 142–155.

[14] J. Anvik, V. Cote, and J. Riehl, "Program wars: a card game for learning programming and cybersecurity concepts," in *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 2019, pp. 393–399.

[15] T. Denning, A. Lerner, A. Shostack, and T. Kohno, "Control-alt-hack: the design and evaluation of a card game for computer security awareness and education," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, 2013, pp. 915–928.

[16] A. T. Forde, "A gamification toolkit for improving cyber security standards adoption," Master's thesis, University of Southampton, September 2020.

[17] "Hacknet," Accessed: 14-12-2020. [Online]. Available: https://hacknet-os.com/

[18] R. Hunicke, M. LeBlanc, and R. Zubek, "Mda: A formal approach to game design and game research," in *Proceedings of the AAAI Workshop on Challenges in Game AI*, vol. 4, no. 1. San Jose, CA, 2004, p. 1722.

[19] "Colour wheel calculator," 2021. [Online]. Available: https://www.sessions.edu/color-calculator/

[20] 2021. [Online]. Available: https://www.w3.org/TR/2018/REC-WCAG21-20180605/

[21] "Colour blindness awareness," 2021. [Online]. Available: https://www.colourblindawareness.org/

[22] D. Nichols, "Coloring for colorblindness," 2021. [Online]. Available: https://davidmathlogic.com/colorblind/#%23E1BE6A-%2340B0A6

[23] 2021. [Online]. Available: https://doodad.dev/pattern-generator/

[24] "Unity 3d," 2021. [Online]. Available: https://unity.com/

[25] "Unity - manual: Getting started with webgl development," 2021. [Online]. Available: https://docs.unity3d.com/Manual/webgl-gettingstarted.html

[26] "Phaser3," 2021. [Online]. Available: https://phaser.io/phaser3

[27] "Photon cloud vs photon server," 2021. [Online]. Available: https://doc.photonengine.com/en-us/realtime/current/getting-started/onpremises-or-saas

[28] "Pun2 documentation," 2021. [Online]. Available: https://doc-api.photonengine.com/en/PUN/v2/index.html

[29] "Mirror networking," 2021. [Online]. Available: https://mirror-networking.gitbook.io/docs/

[30] "Rpcs and raiseevents," 2021. [Online]. Available: https://doc.photonengine.com/en-us/pun/v2/gameplay/rpcsandraiseevent

[31] "Cyberland - Cyber Security Challenge UK," Accessed: 14-12-2020. [Online]. Available: https://www.cybersecuritychallenge.org.uk/what-we-do/schools-programme/cyberland

[32] "Game of Threats," Accessed: 14-12-2020. [Online]. Available: https://www.pwc.co.uk/issues/cyber-security-services/game-of-threats.html

[33] "Webonauts Internet Academy," Accessed: 14-12-2020. [Online]. Available: https://learningworksforkids.com/apps/webonauts-internet-academy/

[34] "Targeted Attack - The Simulation," Accessed: 14-12-2020. [Online]. Available: http://targetedattacks.trendmicro.com/

[35] "Keep Tradition Secure," Accessed: 14-12-2020. [Online]. Available: https://keeptraditionsecure.tamu.edu/

[36] "Cyber Awareness Challenge USA," Accessed: 14-12-2020. [Online]. Available: http://sgschallenge.com/cyber-awareness-challenge/

13 Appendices

13.1 A Review of Serious Cyber Security Games

The following games can be found online at: [31, 32, 33, 34, 11, 10, 35, 17, 36] (in order).

Game	Cyberland - Cyber Security Challenge
Game Type	Web Application Point and Click
Target Audience	Children, Teenagers, Students (High school - University level)
Description	Cyber Security Challenge UK is an organisation which hosts a variety of mini games (Cyberland), competitions and networking between schools, universities, businesses and government institutes
Key Teachings / Findings	Examples of minigames which teach: - Identifying phishing emails - Command line simulator - Firewall simulator (analyse incoming network activity and grant/deny requests - Database simulator -(remove old accounts, sanitise personal information, check admin clearance) - Coffee shop network simulator (using unprotected networks vs VPN and shoulder surfing) - IoT home simulator - making sure all IoT devices have latest software update - Courthouse simulator - demonstrating cyber security laws and ethics - Cipher cracking simulator - Password strength making game - Data leak mystery solver - Malware simulator (demonstrates different types of malware/ransomware and they work)
Mechanics Identified	• Competition • Feedback / Guidance • Tips / Hints • Story • Goals / Objectives

Game	Game of Threats
Game Type	Multiplatform - (Mobile, Tablet, PC), Multiplayer
Target Audience	Businesses - Employees
Description	Employees are split into teams of attackers and defenders who work together to simulate scenarios of cyber attacks and appropriate responses
Key Teachings / Findings	- Teaches people about cyber security trends and to understand the consequences of cyber attacks and what you can do to mitigate the impacts - Helps people understand the mindset of both attackers and defenders- - Prompts discussion with colleagues in teams to popularise cyber security readiness
Mechanics Identified	● Feedback / Guidance ● Incentives / Rewards ● Competition

Game	Webonauts Internet Academy
Game Type	Web Application Point and Click Side Scroller
Target Audience	Children (aged 7-12)
Description	Puts the player as an astronaut in which they can rank up their status by demonstrating smart and good behaviour
Key Teachings / Findings	Teaches children: - How to be respectful online - How to protect themselves online - Looking for trustful website certificates - Establishing privacy settings on profile - Not giving out and using weak passwords
Mechanics Identified	● Avatar ● Feedback / Guidance ● Tips / Hints ● Badges / Privileges

Game	Targeted Attack
Game Type	Web Application Point and Click
Target Audience	Businesses - Employees
Description	Targeted Attack places you as a CEO in a simulation of business growth and defence from cyber attacks
Key Teachings / Findings	Teaches employees: - Smart and safe decision making - Threat level of different types of cyber attacks and how to mitigate them
Mechanics Identified	● Feedback / Guidance ● Story ● Challenge

Game	Classcraft
Game Type	Web Application, Point and Click, Multiplayer, Productivity - Management
Target Audience	School Students
Description	Classcraft incorporates gamifiication principles through the use of management software to set goals and challenges within a classroom and encourages teamwork between students
Key Teachings / Findings	Teaches employees: - Smart and safe decision making - Threat level of different types of cyber attacks and how to mitigate them
Mechanics Identified	● Avatar ● Leaderboard ● Competition ● Badges / Privileges, ● Feedback / Guidance ● Goals / Objectives ● Incentive / Rewards ● Point Systems

Game	Cyber- security Lab
Game Type	Web Application Point and Click
Target Audience	Businesses - Employees
Description	Allows the player to choose a business they'd like to start and require them to spend defence points in different areas of cyber defence
Key Teachings / Findings	Teaches children via minigames: - how to spot phishing emails - how to construct strong passwords - Simple programming principles
Mechanics Identified	● Avatar ● Achievements ● Progress / Levels ● Point System ● Tips / Hints ● Feedback / Guidance

Game	Keep Tradition Secure
Game Type	Web Application Point and Click
Target Audience	University Students
Description	You are a campus student trying to take down a fictional cyber criminal by making smart cyber security decisions
Key Teachings / Findings	Teaches students: - Smart decision making on campus (using public networks vs campus VPN) - Quiz based - Gives out prizes for student participants
Mechanics Identified	● Tips / Hints ● Feedback / Guidance ● Rewards / Incentives

Game	Hacknet
Game Type	Downloadable, Single Player, Point and Click
Target Audience	Gamers
Description	Hacknet is a paid game (on Steam) which is a terminal-based hacking simulator
Key Teachings / Findings	Teaches player: - How to navigate networks - Search for hidden files/folders - Authorisation bypass - Heavy use of terminal/linux commands in a tutorial environment
Mechanics Identified	● Story ● Progress / Levels ● Feedback / Guidance ● Steam Achievements

Game	Cyber Awareness Challenge
Game Type	Downloadable Training Simulator
Target Audience	Businesses Employees
Description	Single Player simulation of everyday life within the workplace and how to behave safely and responsibly
Key Teachings / Findings	- Teaches employees how to be safe in the workplace - Gives points for correct answers and guidance for both right and wrong answers
Mechanics Identified	● Tips / Hints ● Feedback / Guidance ● Story ● Points System